CLAIMS:

1. (Currently Amended): A method for deploying a set of coupled data transformation modules describing a data transformation, the data transformation for transforming a data structure from a first format to a second format, the method comprising the steps of:

receiving an instruction for selecting the set of transformation modules from a memory;

converting each of the set of transformation modules to a common model format, the set of modules having ~~at least one~~ a first transformation module being ~~of a module type of a type set including~~ a language constructed ~~modules~~ module and a second transformation module being a visually constructed ~~modules~~ module, wherein one of the first module or the second module references the other of the first module or the second module; and

generating an executable version of the converted transformation modules suitable for execution by a data transformation engine;

wherein the executable version when executed transforms the data structure from the first format to the second format.

2. (Original): The method of claim 1 further comprising the step of employing a user interface to generate the instruction for coordinating the creation of the converted common models.

3. (Original): The method of claim 2 further comprising the step of updating a module registry to include entries corresponding to each of the converted common modules.

4. (Original): The method of claim 3 further comprising the step of including a name of each of the converted common modules in the entries of the registry, the names for retrieving the corresponding common modules from the memory in response to the instruction.

5. (Original): The method of claim 2, wherein the set of coupled transformation modules includes at least two transformation modules.

6. (Original): The method of claim 5, wherein the executable version is represented by at least one deployment module.

7. (Original): The method of claim 1, wherein the common model format contains all information for use in implementing the transformation functionality of the original coupled transformation modules.

8. (Original): The method of claim 7 further comprising the step of removing a portion of visual interface contents from each of the visually constructed modules during conversion of the visual modules to the common model format.

9. (Original): The method of claim 7, wherein the common model format is different from both the format of the language constructed modules and the format of the visually constructed modules.

10. (Original): The method of claim 9, wherein the common model format is generic for suitable generation of the executable version for a selected one of a plurality of runtime environments for the data transformation engine.

11. (Currently Amended): The method of claim 1, ~~wherein the first step in generation of the executable version is the conversion step, such that the set of coupled transformation modules are reconfigured~~ further comprising reconfiguring a given transformation module directly into the executable version of the coupled transformation module.

12. (Original): The method of claim 1, wherein the step of receiving the instruction is performed after the step of converting the set of transformation modules to a common model format.

13. (Currently Amended): A system for deploying a set of coupled data transformation modules describing a data transformation, the data transformation for transforming a data structure from a first format to a second format, the system comprising:

a memory for storing the set of transformation modules;

a format module for converting each of the set of transformation modules to a common model format, the set of modules having ~~at least one~~ a first transformation module being ~~of a module type of a type set including~~ a language constructed ~~modules~~ module and a second transformation module being a visually constructed ~~modules~~ module, wherein one of the first module or the second module references the other of the first module or the second module; and

a deployment engine for receiving an instruction to select the set of converted transformation modules and for generating an executable version of the converted transformation modules suitable for execution by a data transformation engine;

wherein the executable version when executed transforms the data structure from the first format to the second format.

14. (Original): The system of claim 13 further comprising a user interface to generate the instruction for coordinating the creation of the converted common models.

15. (Original): The system of claim 14 further comprising a module registry for including entries corresponding to each of the converted common modules.

16. (Original): The system of claim 15 further comprising a name of each of the converted common modules included in the entries of the registry, the names for retrieving the corresponding common modules from the memory in response to the instruction.

17. (Original): The system of claim 14, wherein the set of coupled transformation modules includes at least two transformation modules.

18. (Original): The system of claim 17, wherein the executable version is represented by at least one deployment module.

19. (Currently Amended): The system of claim [[1]] 13, wherein the common model format contains all information for use in implementing the transformation functionality of the original coupled transformation modules.

20. (Original): The system of claim 19, wherein a portion of visual interface contents is removed from each of the visually constructed modules during conversion of the visual modules to the common model format.

21. (Original): The system of claim 19, wherein the common model format is different from both the format of the language constructed modules and the format of the visually constructed modules.

22. (Original): The system of claim 21, wherein the common model format is generic for suitable generation of the executable version for a selected one of a plurality of runtime environments for the data transformation engine.

23. (Currently Amended): The system of claim 13, ~~wherein the first step in generation of the executable version is the conversion process, such that the set of coupled transformation modules are reconfigured~~ further comprising reconfiguring a given transformation module directly into the executable version of the coupled transformation module.

24. (Original): The system of claim 13, wherein receiving the instruction is performed after converting the set of transformation modules to a common model format.

25. (Currently Amended): A computer program product for deploying a set of coupled data transformation modules describing a data transformation, the data transformation for

transforming a data structure from a first format to a second format, the computer program product comprising:

a computer readable medium;

a format module stored on the medium for converting each of the set of transformation modules to a common model format, the set of modules having ~~at least one~~ a first transformation module being ~~of a module type of a type set including~~ a language constructed ~~modules~~ module and a second transformation module being a visually constructed ~~modules~~ module, wherein one of the first module or the second module references the other of the first module or the second module; and

a deployment engine module coupled to the format module for receiving an instruction to select the set of converted transformation modules from a memory and for generating an executable version of the converted transformation modules suitable for execution by a data transformation engine;

wherein the executable version when executed transforms the data structure from the first format to the second format.

26. (Currently Amended): A computer readable medium containing computer executable code for deploying a set of coupled data transformation modules describing a data transformation, the data transformation for transforming a data structure from a first format to a second format, the code comprising the steps of:

receiving an instruction for selecting the set of transformation modules from a memory;

converting each of the set of transformation modules to a common model format, the set of modules having ~~at least one~~ a first transformation module being ~~of a module type of a type set including~~ a language constructed ~~modules~~ module and a second transformation module being a visually constructed ~~modules~~ module, wherein one of the first module or the second module references the other of the first module or the second module; and

generating an executable version of the converted transformation modules suitable for execution by a data transformation engine;

wherein the executable version when executed transforms the data structure from the first format to the second format.